CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION, CFSTI
INPUT SECTION 410.11

LIMITATIONS IN REPRODUCTION QUALITY OF TECHNICAL ABSTRACT BULLETIN DOCUMENTS, DEFENSE DOCUMENTATION CENTER (DDC)

AD 615884

☒ 1. AVAILABLE ONLY FOR REFERENCE USE AT DDC FIELD SERVICES. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 2. AVAILABLE COPY WILL NOT PERMIT FULLY LEGIBLE REPRODUCTION. REPRODUCTION WILL BE MADE IF REQUESTED BY USERS OF DDC.

☐ A. COPY IS AVAILABLE FOR PUBLIC SALE.

☐ B. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 3. LIMITED NUMBER OF COPIES CONTAINING COLOR OTHER THAN BLACK AND WHITE ARE AVAILABLE UNTIL STOCK IS EXHAUSTED. REPRODUCTIONS WILL BE MADE IN BLACK AND WHITE ONLY.

TSL-121-2/65

DATE PROCESSED: 16 June 1965

PROCESSOR: F. A. [illegible]

AN OUT-OF-KILTER METHOD FOR MINIMAL COST FLOW PROBLEMS

D. R. Fulkerson

Mathematics Division
The RAND Corporation

P-1825

21 October 1959
Revised 5 April 1960

Reproduced by

The RAND Corporation • Santa Monica • California

The views expressed in this paper are not necessarily those of the Corporation

SUMMARY

A method of solving minimal cost network flow problems is described. This method begins with any circulation, feasible or not, and an arbitrary pricing vector. A labeling procedure is then used to adjust 'out-of-kilter' arcs, that is, arcs that fail to satisfy the optimality properties. It is shown that the method terminates in a finite number of steps, and that in so doing, the status of no arc of the network (as measured by certain 'kilter numbers') is worsened at any step.()

R-ii

AN OUT-OF-KILTER METHOD FOR MINIMAL COST FLOW PROBLEMS[1]

D. R. Fulkerson

1. Introduction In this paper a method of solving minimal cost network flow problems is described and shown to have a certain desirable monotone behavior. The method begins with an arbitrary flow, feasible or not, together with an arbitrary pricing vector, and then uses a labeling procedure to adjust an arc of the network that fails to satisfy the appropriate optimality properties.

To present the basic notions underlying the method, let us consider, for a moment, a general linear program of the form

$$(1.1) \qquad \sum_{j=1}^{n} a_{ij}x_j = b_i \qquad (i = 1, \ldots, m)$$

$$(1.2) \qquad \ell_j \leq x_j \leq u_j \qquad (j = 1, \ldots, n)$$

$$(1.3) \qquad \text{minimize} \sum_{j=1}^{n} c_j x_j .$$

Here the a_{ij}, b_i, ℓ_j, u_j, c_j are given. Now suppose that $x = (x_1, \ldots, x_n)$ is a vector satisfying (1.1) and (1.2), that is, x is feasible, and that there is a dual (or pricing)

[1] This research was supported in part by the Pacific Lighting System.

vector $\pi = (\pi_1, \ldots, \pi_m)$ such that the implications

$$(1.4) \qquad c_j + \sum_{i=1}^{m} \pi_i a_{ij} > 0 \longrightarrow x_j = \ell_j$$

$$(1.5) \qquad c_j + \sum_{i=1}^{m} \pi_i a_{ij} < 0 \longrightarrow x_j = u_j$$

hold for all j. Then it follows that x is a minimizing solution, and thus (1.4), (1.5) might be termed optimality properties.

For a given x satisfying (1.1) and for any π, the following case classification for the j-th component of the program is exclusive and exhaustive:

$$\begin{array}{lll}
(\alpha) & c_j + \sum_i \pi_i a_{ij} > 0, & x_j = \ell_j \\
(\beta) & c_j + \sum_i \pi_i a_{ij} = 0, & \ell_j \leq x_j \leq u_j \\
(\gamma) & c_j + \sum_i \pi_i a_{ij} < 0, & x_j = u_j \\
(\alpha_1) & c_j + \sum_i \pi_i a_{ij} > 0, & x_j < \ell_j \\
(\beta_1) & c_j + \sum_i \pi_i a_{ij} = 0, & x_j < \ell_j \\
(\gamma_1) & c_j + \sum_i \pi_i a_{ij} < 0, & x_j < u_j \\
(\alpha_2) & c_j + \sum_i \pi_i a_{ij} > 0, & x_j > \ell_j \\
(\beta_2) & c_j + \sum_i \pi_i a_{ij} = 0, & x_j > u_j \\
(\gamma_2) & c_j + \sum_i \pi_i a_{ij} < 0, & x_j > u_j .
\end{array}$$

If all components are in one of the states α, β, γ, then x is feasible and optimal. We call these the "in-kilter" states, the others "out-of-kilter" states. The algorithm to be presented for network flow problems concentrates on a particular out-of-kilter component, and gradually puts it in kilter. It does this in such a way that all in-kilter components stay in kilter, whereas any other out-of-kilter component either improves or stays the same, in a sense made precise in section 2.

Section 2 provides a description of the special class of linear programs to which the method applies, together with some preliminary discussion. We assume that the given data for the program are integers (or, equivalently, rationals). Then the algorithm, presented in section 3, works with integers throughout. A proof that the algorithm terminates in a finite number of steps, and that in so doing it possesses the monotone property roughly described above, is sketched in section 4. Some comparisons with other methods for solving minimal cost flow problems are made in section 5.

For the particular class of programs being considered, the assumption that the initial x satisfies (1.1) is unimportant, since such an x is immediately available, e.g., $x = 0$. But starting with a good guess for x and π will decrease computation time. One situation for which the present algorithm is particularly appropriate would be in solving a sequence of flow problems, where each problem of the sequence differs only slightly from its predecessor. Then the old optimal x and π could be used to initiate the computation for the new problem.

We should like to express our appreciation to G. B. Dantzig, whose criticism of an earlier version of this paper in which the intitial x was assumed feasible, led us to reconsider the problem from the standpoint of infeasible x.

2. Notation, definitions, and problem description.

We suppose given a network consisting of nodes 1, 2, ..., n together with directed arcs ij (from node i to node j). Each arc ij has associated with it three integers: ℓ_{ij} (the arc lower bound), u_{ij} (the arc upper bound or capacity), and c_{ij} (the arc cost), with $0 \leq \ell_{ij} \leq u_{ij}$.

It is convenient to describe the problem in terms of circulations [12], rather than flows from sources to sinks [7, 8, 9]. By a circulation we shall mean a nonnegative integral vector $x = (x_{ij})$, one component for each arc ij, that satisfies the conservation equations

(2.1) $$\sum_j (x_{ij} - x_{ji}) = 0 \qquad (i = 1, \ldots, n).$$

If the circulation x also satisfies

(2.2) $$\ell_{ij} \leq x_{ij} \leq u_{ij} \qquad \text{(all arcs } ij\text{)},$$

we call x a feasible circulation. We shall refer to a particular component x_{ij} of a circulation as the arc flow x_{ij} or the flow in arc ij.

A feasible circulation x that minimizes the cost form

(2.3) $$\sum_{ij} c_{ij}x_{ij}$$

over all feasible circulations is optimal. The problem we are considering is that of constructing an optimal circulation. Of course feasible circulations may not exist, in which case we want to discover this fact. It is known [12] that a necessary and sufficient condition for the existence of a feasible circulation is that the inequalities

$$(2.4) \qquad \sum_{\substack{i \text{ in } L \\ j \text{ in } \bar{L}}} u_{ij} \geq \sum_{\substack{i \text{ in } L \\ j \text{ in } \bar{L}}} \ell_{ji}$$

hold for all subsets L of nodes. Here $\bar{L}$ denotes the complement of L. The conditions (2.4) are easily shown to be necessary; their sufficiency can be proved in various ways, for example, using the max flow min cut theorem [7, 8] or the supply-demand theorem [11].

Let $\pi = (\pi_i)$ be a vector of integers, one component for each node i. We call π a *pricing vector*, and refer to its components as *node prices*. Optimality properties for the problem are that the implications

$$(2.5) \qquad c_{ij} + \pi_i - \pi_j > 0 \longrightarrow x_{ij} = \ell_{ij}$$

$$(2.6) \qquad c_{ij} + \pi_i - \pi_j < 0 \longrightarrow x_{ij} = u_{ij}$$

hold for all arcs ij. That is, if x'is a feasible circulation, and if there is a pricing vector π such that (2.5), (2.6) hold, then x is optimal. We shall shorten the notation by setting

$$\bar{c}_{ij} = c_{ij} + \pi_i - \pi_j. \tag{2.7}$$

For a given circulation x and pricing vector π, an arc ij is in just one of the following states:

(α) $\bar{c}_{ij} > 0,\ x_{ij} = \ell_{ij}$

(β) $\bar{c}_{ij} = 0,\ \ell_{ij} \leq x_{ij} \leq u_{ij}$

(γ) $\bar{c}_{ij} < 0,\ x_{ij} = u_{ij}$

(α_1) $\bar{c}_{ij} > 0.\ x_{ij} < \ell_{ij}$

(β_1) $\bar{c}_{ij} = 0.\ x_{ij} < \ell_{ij}$

(γ_1) $\bar{c}_{ij} < 0,\ x_{ij} < u_{ij}$

(α_2) $\bar{c}_{ij} > 0,\ x_{ij} > \ell_{ij}$

(β_2) $\bar{c}_{ij} = 0,\ x_{ij} > u_{ij}$

(γ_2) $\bar{c}_{ij} < 0,\ x_{ij} > u_{ij}$.

We say that an arc ij is _in kilter_ if it is in one of the states α, β, γ; otherwise the arc is _out of kilter_. Thus to solve the problem, we need to get all arcs in kilter.

With each state that an arc ij can be in, we shall associate a nonnegative integer, called the _kilter number_ of the arc in the given state. An in-kilter arc has kilter number 0; the arc kilter numbers corresponding to out-of-kilter states are listed below:

(α_1) $\ell_{ij} - x_{ij}$

(β_1) $\ell_{ij} - x_{ij}$

(γ_1) $\bar{c}_{ij}(x_{ij} - u_{ij})$

(α_2) $\bar{c}_{ij}(x_{ij} - \ell_{ij})$

(β_2) $x_{ij} - u_{ij}$

(γ_2) $x_{ij} - u_{ij}$.

Thus out-of-kilter arcs have positive kilter numbers. The kilter numbers for states α_1, β_1, β_2, γ_2 measure infeasibility for the arc flow x_{ij}, while the kilter numbers for states γ_1, α_2 are a measure of the degree to which the optimality properties (2.5), (2.6) fail to be satisfied.

The algorithm stated in the following section has the property that all arc kilter numbers are monotone nonincreasing throughout the computation. However, steps can occur that change no kilter number, and this complicates the proof of termination somewhat.

We need a few other notions before stating the algorithm, the main one being that of a path from some node to another in a network. Let i_1, i_2, ..., i_m be a sequence of distinct nodes of a network such that either $i_k i_{k+1}$ or $i_{k+1} i_k$ is an arc, $k = 1, \ldots, m - 1$. Picking out, for each k, one of these two possibilities, we call the resulting sequence of nodes and arcs a <u>path from</u> i_1 <u>to</u> i_m. Arcs $i_k i_{k+1}$ that belong to the path are <u>forward arcs</u> of the path; arcs $i_{k+1} i_k$ that belong to the path are <u>reverse arcs</u> of the path. If we alter the definition of a path by stipulating that $i_1 = i_m$, we call the resulting sequence of nodes and arcs a <u>cycle</u>.

3. <u>**An out-of-kilter algorithm.**</u> **The algorithm of this section uses a modified labeling procedure [8, 9] as its basic routine. In general, the labeling procedure is a** search for a path (having certain desired properties) from some node to another. We start labeling from a given node, called the <u>origin</u>, attempting to reach some other given node, called the <u>terminal</u>. To initiate the modified procedure, we assign the label $[o, \infty]$ to the origin; the following <u>labeling rules</u> are then applied:

(3.1) If node i is labeled $[k^{\pm}, \epsilon_i]$, node j is unlabeled, and if ij is an arc such that either

(a) $\bar{c}_{ij} > 0$, $x_{ij} < \ell_{ij}$,

(b) $\bar{c}_{ij} \leq 0$, $x_{ij} < u_{ij}$,

then node j receives the label $[i^{+}, \epsilon_j]$, where $\epsilon_j = \min(\epsilon_i, \ell_{ij} - x_{ij})$ in case (a), $\epsilon_j = \min(\epsilon_i, u_{ij} - x_{ij})$ in case (b).

(3.2) If node i is labeled $[k^{\pm}, \epsilon_i]$, node j is unlabeled, and if ji is an arc such that either

(a) $\bar{c}_{ji} \geq 0$, $x_{ji} > \ell_{ji}$,

(b) $\bar{c}_{ji} < 0, \ x_{ji} > u_{ji}$

then node j receives the label $[i^-, \epsilon_j]$, where
$\epsilon_j = \min(\epsilon_i, x_{ji} - \ell_{ji})$ in case (a),
$\epsilon_j = \min(\epsilon_i, x_{ji} - u_{ji})$ in case (b).

Here x is a circulation and π a pricing vector.

The labeling procedure terminates in one of two ways, called <u>breakthrough</u> and <u>nonbreakthrough</u>, respectively: either the terminal receives a label, or no more labels can be assigned and the terminal has not been labeled.

If breakthrough occurs, a path from origin to terminal can be located by backtracking from the terminal, using the first members of the label pairs. If, in this backtracking, a node j is reached that carries the label $[i^+, \epsilon_j]$, then ij is a forward arc of the path from origin to terminal; if j is labeled $[i^-, \epsilon_j]$, then ji is a reverse arc of the path. Thus forward arcs of the path satisfy either (3.1a) or (3.1b), whereas reverse arcs of the path satisfy (3.2a) or (3.2b).

If nonbreakthrough results, we let L and $\bar{L}$ denote the sets of labeled and unlabeled nodes respectively, and define two subsets of arcs:

$$(3.3) \qquad \mathcal{A}_1 = \left\{ ij \mid i \text{ in } L, \ j \text{ in } \bar{L}, \ \bar{c}_{ij} > 0, \ x_{ij} \leq u_{ij} \right\}$$

$$(3.4) \qquad \mathcal{A}_2 = \left\{ ji \mid i \text{ in } L, \ j \text{ in } \bar{L}, \ \bar{c}_{ji} < 0, \ x_{ji} \geq \ell_{ji} \right\}.$$

We then define

$$(3.5)\qquad \delta_1 = \min_{ij \text{ in } \mathcal{A}_1} (\bar{c}_{ij})$$

$$(3.6)\qquad \delta_2 = \min_{ji \text{ in } \mathcal{A}_2} (-\,\bar{c}_{ji})$$

$$(3.7)\qquad \delta = \min(\delta_1,\ \delta_2).$$

Here δ_i $(i = 1, 2)$ is a positive integer or ∞ according as $\mathcal{A}_i$ is nonempty or empty.

The complete algorithm now runs as follows. Start the computation with any circulation x and any pricing vector π. Next locate an out-of-kilter arc st and go on to the appropriate case below:

(α_1) $\bar{c}_{st} > 0$, $x_{st} < \ell_{st}$. The origin for labeling is t, the terminal s. If breakthrough results, add $\epsilon = \min(\epsilon_s, \ell_{st} - x_{st})$ to the flow in all forward arcs of the path from t to s, subtract ϵ from the flow in all reverse arcs, and add ϵ to x_{st}. If nonbreakthrough results, add δ defined in (3.7) to all π_i for i in $\mathcal{L}$.

(β_1) or (γ_1) $\bar{c}_{st} = 0$, $x_{st} < \ell_{st}$, or $\bar{c}_{st} < 0$, $x_{st} < u_{st}$. Same as (α_1), except

$$\epsilon = \min(\epsilon_s, u_{st} - x_{st}).$$

(α_2) or (β_2) $\bar{c}_{st} > 0$, $x_{st} > \ell_{st}$, or $\bar{c}_{st} = 0$, $x_{st} > u_{st}$. The origin for labeling is s, the terminal t. If breakthrough results, add $\epsilon = \min(\epsilon_t, x_{st} - \ell_{st})$ to the flow in all forward arcs of the path from s to t, subtract ϵ from the flow in all reverse arcs, and subtract ϵ from x_{st}. If nonbreakthrough results, add δ defined in (3.7) to all π_i for i in L.

(γ_2) $\bar{c}_{st} < 0$, $x_{st} > u_{st}$. Same as (α_2) or (β_2), except $\epsilon = \min(\epsilon_t, x_{st} - u_{st})$.

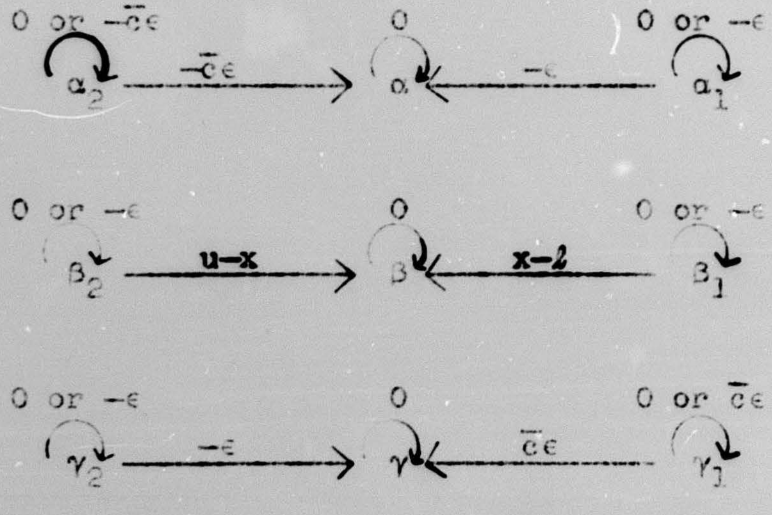
The labeling process is repeated for the arc st until either st is in kilter, or until a nonbreakthrough occurs for which the node price change $\delta = \infty$. In the latter case, stop. (There is no feasible circulation). In the former case, locate another out-of-kilter arc and continue.

4. Termination and the monotone property. Suppose that arc st is out of kilter, say in state α_1. The origin for labeling is t, the terminal s. The arc st cannot be used to label s directly, since neither (3.2a) nor (3.2b) is applicable.

Consequently, if breakthrough occurs, the resulting path from t to s, together with the arc st, is a cycle. Then the flow changes that are made on arcs of this cycle again yield a circulation. Moreover, the labeling rules have been selected in such a way that kilter numbers for arcs of this cycle do not increase, and at least one, namely, for arc st, decreases. Kilter numbers for arcs not in the cycle of course don't change.

Similar remarks apply if st is in one of the other out-of-kilter states.

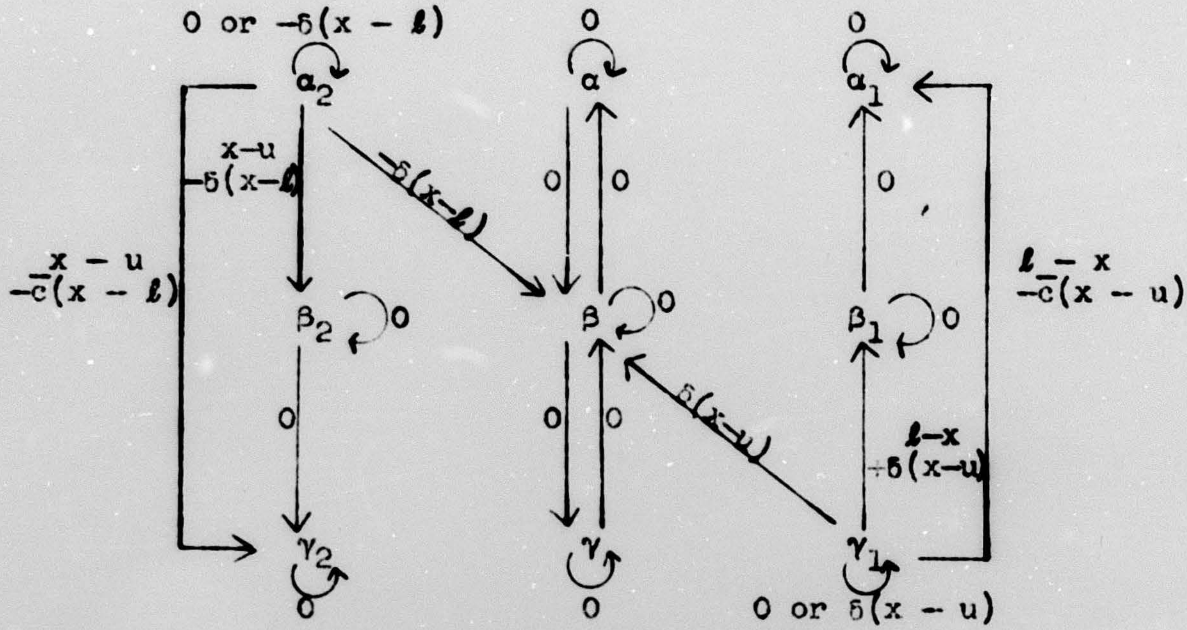
We summarize the possible effects of a breakthrough on an arc ij in the diagram below, which shows the state transitions that may occur following breakthrough. If a transition is possible, the number recorded beside the corresponding arrow represents the change in kilter number. (The subscripts ij are omitted in the diagram.)



Breakthrough Diagram

Verification of the breakthrough diagram is straightforward. For example, suppose arc ij is in state α_2, with $\bar{c}_{ij} > 0$, $x_{ij} > \ell_{ij}$, and kilter number $\bar{c}_{ij}(x_{ij} - \ell_{ij}) > 0$. If ij is not an arc of the cycle of flow changes, then ij remains in state α_2 with zero change in kilter number. If the flow in arc ij has changed as a result of the breakthrough, then either ij is the arc st or, by the labeling rules (3.1), (3.2), ij is a reverse arc of the path from origin to terminal. Specifically, i was labeled from j using (3.2a). In either case, x_{ij} decreases by the positive integer $\epsilon \leq x_{ij} - \ell_{ij}$, the new state for ij is α_2 or α, and hence the kilter number for ij has decreased by $\bar{c}_{ij}\ \epsilon > 0$. The rest of the diagram may be verified similarly.

The state transitions and changes in kilter number that may occur following a nonbreakthrough with $\delta < \infty$ are indicated in the following diagram. (Again the subscripts ij are omitted.)



<u>Nonbreakthrough diagram</u>

Again we omit a detailed verification, but consider, for example, an arc ij in state γ_1, so that $\bar{c}_{ij} < 0$, $x_{ij} < u_{ij}$, having kilter number $\bar{c}_{ij}(x_{ij} - u_{ij}) > 0$ before the node price change is made. If both i and j are in L or both in $\bar{L}$, then $\bar{c}_{ij}$ remains the same after the node price change, and consequently ij stays in state γ_1 with no change in kilter number. We cannot have i in L, j in $\bar{L}$ (labeling rule (3.1b)), and hence the remaining possibility is i in $\bar{L}$, j in L. Then $\bar{c}_{ij}$ is increased by $\delta > 0$. Consequently the arc ij either remains in state γ_1, (if $\delta < -\bar{c}_{ij}$), goes into state β (if $\delta = -\bar{c}_{ij}$ and $x_{ij} \geq \ell_{ij}$), into state β_1 (if $\delta = -\bar{c}_{ij}$ and $x_{ij} < \ell_{ij}$), or into state α_1 (if $\delta > -\bar{c}_{ij}$ and $x_{ij} < \ell_{ij}$), and the corresponding changes in kilter number are respectively $\delta(x_{ij} - u_{ij}) < 0$, $\delta(x_{ij} - u_{ij}) < 0$, $\ell_{ij} - x_{ij} + \delta(x_{ij} - u_{ij}) \leq 0$, $\ell_{ij} - x_{ij} - \bar{c}_{ij}(x_{ij} - u_{ij}) \leq 0$. (The remaining logical possibility $\delta > -\bar{c}_{ij}$, $x_{ij} \geq \ell_{ij}$ cannot occur, since if $x_{ij} \geq \ell_{ij}$, then ij is in $\mathcal{A}_2$ defined by (3.4), and hence $\delta \leq -\bar{c}_{ij}$.)

It follows from the breakthrough and nonbreakthrough diagrams that kilter numbers are monotone nonincreasing throughout the computation. Moreover, if breakthrough occurs, at least one arc kilter number decreases. Thus to prove that the algorithm terminates, it suffices to show that an infinite sequence of successive nonbreakthroughs, each with $\delta < \infty$, is impossible. To show this, let us suppose that a labeling resulting in nonbreakthrough with $\delta < \infty$ has occurred, and let L, $\bar{L}$ denote the labeled and unlabeled sets of nodes. After changing node prices, the new $\bar{c}$ vector, which we denote by $\bar{c}'$, has components given in terms of the old by

$$(4.1) \qquad \bar{c}'_{ij} = \begin{cases} \bar{c}_{ij} - \delta & \text{for i in L, j in } \bar{L} \\ \bar{c}_{ij} + \delta & \text{for i in } \bar{L}\text{, j in L} \\ \bar{c}_{ij} & \text{otherwise.} \end{cases}$$

If the arc st is still out of kilter, then the origin is the same for the next labeling, and it follows from (4.1) and the labeling rules that every node of L may again be labeled. Thus if the new labeling again results in nonbreakthrough with labeled set L', we have $L \subseteq L'$. Let $\mathcal{A}_1'$, $\mathcal{A}_2'$ denote the new sets defined in terms of L', $\bar{c}'$ (and x) by (3.3), (3.4), and suppose $L = L'$. Then, from (4.1) we have $\mathcal{A}_1' \subseteq \mathcal{A}_1$, $\mathcal{A}_2' \subseteq \mathcal{A}_2$, and at least one of these inclusions is proper by (3.5), (3.6), (3.7). Hence the new labeling either assigns a label to at least one more node, or failing this, an arc is removed from one of the sets $\mathcal{A}_1$ or $\mathcal{A}_2$. It follows that, after finitely many nonbreakthroughs with $\delta < \infty$, we either get the arc st in kilter, obtain a breakthrough, or obtain a nonbreakthrough with $\delta = \infty$.

If a nonbreakthrough with $\delta = \infty$ occurs, then there is no feasible circulation. For if $\delta = \infty$, it follows from (3.3), (3.4) and the labeling rules (3.1), (3.2) that $x_{ij} \geq u_{ij}$ for i in L, j in $\bar{L}$, and $x_{ji} \leq \ell_{ji}$ for i in L, j in $\bar{L}$. Moreover, for the arc st, either t is in L, s in $\bar{L}$ with $x_{st} < \ell_{st}$, or s is in L, t in $\bar{L}$ with $x_{st} > u_{st}$. (This is immediate for cases α_1, β_1, β_2, γ_2 of the algorithm, and follows from (3.3) and the assumption $\delta = \infty$ for case α_2, from (3.4) and the assumption $\delta = \infty$ for case γ_1.) Hence, summing the equations (2.1) over i in L and noting cancellations, we obtain in all cases

$$0 = \sum_{\substack{i \text{ in } L \\ j \text{ in } \bar{L}}} (x_{ij} - x_{ji}) > \sum_{\substack{i \text{ in } L \\ j \text{ in } \bar{L}}} (u_{ij} - \ell_{ji}).$$

But this violates the feasibility condition (2.4). Thus $\delta = \infty$ implies there is no feasible circulation.

To sum up, the algorithm terminates after finitely many applications of the labeling procedure, either with all arcs in kilter (in which case the feasible circulation is optimal), or with the conclusion that there is no feasible circulation. Moreover, all arc kilter numbers are monotone nonincreasing throughout the computation.

It is worthwhile to note the simplification that occurs if the method of the preceding section is initiated with a feasible circulation. The states α_1, β_1, β_2, γ_2 are then empty to begin with, and consequently remain empty throughout the computation. Hence at each nonbreakthrough (as well as each breakthrough), the kilter number for at least one arc, namely st, decreases by a positive integer. In many minimal cost flow problems, a starting feasible circulation is readily at hand. For example, in the Hitchcock problem [1, 2, 3, 4] or the assignment problem [5, 6, 10], such is the case.

5. Comparison with other methods. The method of section 3 is a generalization of the method of [9] for solving minimal cost flow problems, which in itself generalizes the methods of [5, 6, 8, 10] for solving Hitchcock and assignment problems. In [9] the fundamental problem was that of finding a maximal feasible flow from source node 1 to sink node n that minimizes cost over all such flows. (Also the lower bounds were assumed zero on all arcs. This is not really a restriction, since a change of variables will accomplish this, if desired.) If we add to the network the special arc n1 with $\ell_{n1} = 0$, $u_{n1} = U$,

$c_{n1} = - C$ (U and C large), and consider feasible circulations in the enlarged network, then the method of section 3 is applicable to such problems. Or if it is desired to find an optimal flow from 1 to n of given value $v = \sum_j (x_{1j} - x_{j1})$ in the original network, we can add the arc n1 with $\ell_{n1} = u_{n1} = v$, $c_{n1} = 0$, in order to cast the problem in circulation form.

The method of [9] begins with the zero flow from source 1 to sink n (which satisfies the bounds on arc flows because lower bounds are zero), and all node prices zero. It was also assumed that the given arc costs are nonnegative. Equivalently, if we take $\ell_{n1} = 0$, $u_{n1} = U$, $c_{n1} = - C$ and begin the algorithm of section 3 with the zero circulation and all node prices zero, then the special arc n1 is the only out-of-kilter arc (it is in state γ_1), and hence it remains the only out-of-kilter arc throughout the computation. Then the method of section 3 reduces to that of [9].

It is also informative to note some of the major contrasts between this method and the simplex method [4] for solving such problems. First of all, the simplex method would be done in two phases, the first phase being a search for a feasible circulation, the second for an optimal circulation. (Throughout both of these phases, the simplex method would work with basic solutions, a concept that plays no role in this method.) Here we have combined the two phases. Ignoring this difference, however, and assuming that both methods start with a feasible circulation, the main contrast, apart from mechanics of operation, appears to lie in the fact that, for the simplex method, the kilter numbers are not monotone. For example, arcs that were in kilter at some stage of the simplex computation can go out of kilter at later stages.

REFERENCES

1. Hitchcock, F. L., 'The Distribution of a Product from Several Sources to Numerous Localities,' *Jour. Math. and Physics*, 2 (1941) 224–230.

2. Kantorovitch, L., 'On the Translocation of Masses, *Dokl. Akad. Nauk S.S.R.* 37 (1942) 199–201.

3. Koopmans, T. C., and S. Reiter, A Model of Transportation,' *Cowles Commission Monograph 13*, 1951.

4. Dantzig, G. B., 'Application of the Simplex Method to a Transportation Problem, *Cowles Commission Monograph 13*, 1951.

5. Kuhn, H. W., 'The Hungarian Method for Solving the Assignment Problem, *Naval Res. Logist. Quart.* 2 (1955) 83–97.

6. Kuhn, H. W., 'Variants of the Hungarian Method for Assignment Problems,' *Naval Res. Logist. Quart.* 3 (1956) 253–258.

7. Ford, L. R., Jr., and D. R. Fulkerson, 'Maximal Flow Through a Network,' *Can. Jour. Math.* 8 (1956) 399–404.

8. Ford, L. R., Jr., and D. R. Fulkerson, A Simple Algorithm for finding Maximal Netwc [illegible] ow and an Application to the Hitchcock Problem," *Can.* [illegible] *Math.* 9 (1957) 210–218.

9. Ford, L. R., Jr., and D. R. Fulkerson, 'Constructing Maximal Dynamic Flows from Static Flows, *Operations Research* 6 (1958) 419–433.

10. Munkres, J., Algorithms for the Assignment and Transportation Problems, *Jour. of the Soc. for Ind. and Appl. Math.* 5 (1957) 32–38.

11. Gale, D., A Theorem on Flows in Networks, *Pacific Jour. Math.* 7 (1957) 1073–1082.

12. Hoffman, A. J., "Some Recent Applications of the Theory of Linear Inequalities to Extremal Combinatorial Analysis," to appear in the Amer. Math. Soc. publication of the symposium on combinatorial designs and analysis.